Exercice 1 :

```c
#include <stdio.h>

// Déclaration de la structure Compte
struct Compte {
    char nom[50];
    int numeroCompte;
    float montant;
};

int main() {
    // Déclaration de compte1 et compte2 de type Compte
    struct Compte compte1;
    struct Compte compte2;

    struct Compte tabCompteEtudiant[20];
    //Saisie des informations des 20 étudiants
    for (int i = 0; i < 20; i++) {
        // Saisie des informations pour chaque étudiant
        printf("Saisir le nom de l'etudiant %d: ", i + 1);
        scanf("%s", tabCompteEtudiant[i].nom);

        printf("Saisir le numero de compte de l'etudiant %d: ", i + 1);
        scanf("%d", tabCompteEtudiant[i].numeroCompte);

        printf("Saisir le montant du compte de l'etudiant %d: ", i + 1);
        scanf("%f", tabCompteEtudiant[i].montant);
    }

    // Vous pouvez utiliser les données ainsi collectées selon vos besoins.

    return 0;
}
```

Exercice 2 :

```c
#include <stdio.h>
#include <string.h>

// Déclaration de la structure Employe
struct Employe {
    char NSS[15];
    char Nom[50];
    char Prenom[50];
    char Division[50];
    float Salaire;
};

int main() {
    int N;
    float moyenneSalaires = 0;
    float moyenneSalairesPersonnels = 0;

    // Saisir le nombre d'employés N
    printf("Saisir le nombre d'employes : ");
    scanf("%d", &N);

    // Déclaration d'un tableau d'Employe
    struct Employe tabEmployes[N];

    for (int i = 0; i < N; i++) {
        // Saisie des informations pour chaque employé
        printf("\nSaisir NSS de l'employe %d: ", i + 1);
        scanf("%s", tabEmployes[i].NSS);

        printf("Saisir Nom de l'employe %d: ", i + 1);
        scanf("%s", tabEmployes[i].Nom);
```

```
        printf("Saisir Prenom de l'employe %d: ", i + 1);
        scanf("%s", tabEmployes[i].Prenom);

        printf("Saisir Division de l'employe %d: ", i + 1);
        scanf("%s", tabEmployes[i].Division);

        printf("Saisir Salaire de l'employe %d: ", i + 1);
        scanf("%f", &tabEmployes[i].Salaire);

        // Ajouter le salaire à la moyenne générale
        moyenneSalaires += tabEmployes[i].Salaire;

        // Si la division est "Personnels", ajouter le salaire à la moyenne
spécifique
        if (strcmp(tabEmployes[i].Division, "Personnels") == 0) {
            moyenneSalairesPersonnels += tabEmployes[i].Salaire;
        }
    }

    // Calculer la moyenne des salaires générale
    if (N != 0) {
        moyenneSalaires /= N;
        printf("\nLa moyenne des salaires de tous les employes est : %.2f\n",
moyenneSalaires);

        // Calculer la moyenne des salaires de la division "Personnels"
        moyenneSalairesPersonnels /= N;
        printf("La moyenne des salaires des employes de la division 'Personnels'
est : %.2f\n",
               moyenneSalairesPersonnels);
    } else {
        printf("\nAucun employe saisi.\n");
    }

    return 0;
}
```

Exercice 3 :

```
#include <stdio.h>

// Définition du type de données Athlete
struct Athlete {
    char nom[50];
    char prenom[50];
    int age;
    float essais[3];
};

// Fonction pour trouver la meilleure longueur pour un athlète
float meilleureLongueur(struct Athlete athlete, int *numeroEssai) {
    float meilleure = athlete.essais[0];
    *numeroEssai = 1;

    for (int i = 1; i < 3; i++) {
        if (athlete.essais[i] > meilleure) {
            meilleure = athlete.essais[i];
            *numeroEssai = i + 1;
        }
    }

    return meilleure;
}

int main() {
    const int N = 3;  // N constante
```

```c
    // Déclaration d'un tableau d'Athlete
    struct Athlete participants[N];

    // Saisie des données pour chaque athlète
    for (int i = 0; i < N; i++) {
        printf("\nSaisir le nom de l'athlete %d: ", i + 1);
        scanf("%s", participants[i].nom);

        printf("Saisir le prenom de l'athlete %d: ", i + 1);
        scanf("%s", participants[i].prenom);

        printf("Saisir l'age de l'athlete %d: ", i + 1);
        scanf("%d", &participants[i].age);

        for (int j = 0; j < 3; j++) {
            printf("Saisir la longueur de saut de l'athlete %d, essai %d: ", i + 1,
j + 1);
            scanf("%f", &participants[i].essais[j]);
        }
    }

    // Affichage de la meilleure longueur pour chaque athlète
    for (int i = 0; i < N; i++) {
        int numeroEssai;
        float meilleure = meilleureLongueur(participants[i], &numeroEssai);

        // Affichage du résultat
        printf("%s %s %.2f m %d.essai\n", participants[i].nom,
participants[i].prenom, meilleure, numeroEssai);
    }

    return 0;
}
```

Exercice 4 :

```c
#include <stdio.h>

// Définition de la structure Tetud_info
struct Tetud_info {
    char nom[50];
    char prenom[50];
    float notes[3];
    float moyenne;
};

int main() {
    const int N = 3;  // N constant, nombre d'étudiants

    // Déclaration d'un tableau de structures Tetud_info
    struct Tetud_info listeEtudiants[N];

    // Saisir les informations de tous les étudiants
    for (int i = 0; i < N; i++) {
        printf("\nEtudiant %d\n", i + 1);

        // Saisir le nom et le prénom
        printf("Saisir le nom de l'etudiant : ");
        scanf("%s", listeEtudiants[i].nom);

        printf("Saisir le prenom de l'etudiant : ");
        scanf("%s", listeEtudiants[i].prenom);

        // Saisir les notes des trois devoirs
        for (int j = 0; j < 3; j++) {
            printf("Saisir la note du devoir %d : ", j + 1);
            scanf("%f", &listeEtudiants[i].notes[j]);
        }
```

```
        // Calculer la moyenne
        listeEtudiants[i].moyenne =
                (listeEtudiants[i].notes[0] + listeEtudiants[i].notes[1] +
listeEtudiants[i].notes[2]) / 3.0;
    }

    // Afficher tous les étudiants avec une moyenne >= 10
    printf("\nEtudiants avec une moyenne >= 10 :\n");
    for (int i = 0; i < N; i++) {
        if (listeEtudiants[i].moyenne >= 10) {
            printf("Nom: %s Prenom: %s Moyenne: %.2f\n", listeEtudiants[i].nom,
listeEtudiants[i].prenom,
                   listeEtudiants[i].moyenne);
        }
    }

    return 0;
}
```